一般社団法人　電子情報通信学会
THE INSTITUTE OF ELECTRONICS,
INFORMATION AND COMMUNICATION ENGINEERS

信学技報
IEICE Technical Report
NS2021-78(2021-10)

［依頼講演］　デュアルチャネルIP・NDN変換ゲートウェイのスループット解析

フェリファリアント† 　上山　憲昭††

† 福岡大学 大学院 工学研究科 電子情報工学専攻　〒 814–0180 福岡市城南区七隈 8–19–1
†† 立命館大学 情報理工学部　〒 525–8577 滋賀県草津市野路東 1–1–1
E-mail: †td196502@cis.fukuoka.ac.jp, ††kamiaki@fc.ritsumei.ac.jp

あらまし　コンテンツを効率的に転送できる新しいネットワークアーキテクチャとして情報指向ネットワーク (NDN: named-data networking) が注目されている．しかしネットワーク上のすべての IP ルータが一度に NDN ルータに置き換わることは考えにくく，両方のタイプのルータが混在した環境を想定する必要がある．デュアルチャネル変換ゲートウェイは，2 つの異なるチャネルを用いて，Interest パケットとデータパケットを区別することで，ユーザのプライバシィを維持したまま IP パケットと NDN パケットとの間のプロトコル変換を可能とする．本稿では，簡易な解析モデルを用いて，デュアルチャネルゲートウェイのスループットを導出し，エミュレータの結果と比較する．その結果，解析モデルは平均約 70 パーセントの精度でゲートウェイスループットを予測できることを示す．.

キーワード　NDN，マイグレーション，スループット解析

[Invited Lecture] Throughput Analysis of Translation Gateway Between IP and NDN Using Dual Channel

Feri FAHRIANTO† and Noriaki KAMIYAMA††

† Graduate School of Engineering, Fukuoka University
8–19–1, Nanakuma, Jounan, Fukuoka 814–0180
†† College of Information Science and Engineering, Ritsumeikan University
1–1–1 Nojihigashi, Kusatsu, Shiga 525–0058
E-mail: †td196502@cis.fukuoka.ac.jp, ††kamiaki@fc.ritsumei.ac.jp

Abstract　Named-Data Networking (NDN) is expected to be a new network architecture which efficiently transmitts content items. However, it is unrealistic to assume that all IP routers will be replaced by NDN routers at the same time, so we need to consider the environment in which both IP routers and NDN routers exist in networks. The dual-channel translation gateway can transform between IP packet and NDN packet without violating the user privacy by taging and differentiating interest packets and data packets using two channels. The gateway offers a means to transform the IP to the NDN protocol back and forth utilizing two separate channels devoted to the interest and the data packet. As a result, the IP protocol may connect and interact with the NDN protocol. The throughput of the dual-channel gateway is examined in this study. It describes the relationship between the content hit ratio and throughput performance in the gateway. It also displays the mathematical throughput model and compares it to the emulator results. The finding reveals that the analysis model can predict the gateway throughput with accuracy on average about 70 % from the emulator.

Key words　NDN, migration, throughput analysis

1. Introduction

Information-Centric Networking (ICN) changes the data communication paradigm by naming the data object in the network. The ICN data transaction necessitates the use of two distinct packets, namely the interest and data packets.

To get content from a certain producer, the consumer sends an interest packet containing a prefix name in advance, and the producer responds with a data packet. The ICN communication primitive operates in a pull-driven manner differing from the IP that operates alternatively in push-based or pull-based between the host.

As an active growing ICN technology, the Named Data Networking (NDN) has a rapid implementation in the current communication network. In the future, NDN technology deployment will finally dominate the global communication network. During the transition phase, however, the coexistence of IP and NDN will be unavoidable. Therefore, the dual-channel IP-to-NDN translation gateway, a translation gateway that enables both protocols to transform back and forth between the IP and NDN protocol, can ensure a smooth transition during the migration.

The encapsulation gateway has provided a generic IP-to-NDN migration, but it works at the application layer that creates overhead in packet size. Moreover, the prefix name carried inside the IP payload may create a potential security issue in the network, like the man-in-the-middle attack. The hICN [4] can semantically translate the ICN behaviour inside the IP packet header using the concatenation of IP address and port number. However, hICN prefix name length is limited to the length of IP address and port number that causes a problem for long assigning prefix names. Hence, the dual-channel is designed to eradicate these problems.

We discuss the translation method in terms of how to introduce the NDN behaviour into the IP entity in order to semantically replicate the NDN behaviour. As a result, we purposefully divide the channel into two sorts of packets: interest and data channel. By implementing these channels, the gateway may detect the IP packet labelled as an interest packet or a data packet in the network layer. The asymmetric name resolution table is provided to provide a naming service for binding an IP address to a variable-length prefix name that is statically set in advance. The name resolution table is made up of two separate tables: the producer table and the consumer table.

This article describes the dual-channel IP-to-NDN translation gateway that enables the IP protocol to imitate the NDN protocol behaviour. As a result, the IP endpoint can leverage its semantic from the IP to the NDN. Furthermore, a throughput model is analyzed to investigate the correlation between cache size and throughput. We compare the model with the emulator to find the accuracy of the throughput model. This paper is organized into six sections: introduction, related work, dual-channel translation gateway, throughput analysis model, numerical evaluation, conclusion, and future work.

2. Related work

Several designs have been written and recommended for the IP and NDN co-existence. The extreme suggestion is to redevelop all IP applications with NDN-friendly protocols, which requires enormous cost and time. G. Carofiglio et al. proposed, hICN, the partial migration schemes where the ICN/NDN packet was inserted in IP packet [4]. The prefix name was constructed from a combination of IP and TCP/UDP headers that resulted in an encoded prefix name. This scheme can forward the NDN packet inside the IP network, but it requires a pair of hICN routers to extract the hidden NDN packet from the IP packet. Moreover, the hICN has a limited length of the prefix name that causes a problem for the longer and various prefix names.

Moiseenko et al. proposed a TCP/ICN proxy pair that can carry TCP traffic over an ICN infrastructure, namely forward proxy and reverse proxy [8]. The proxy approach managed to work correctly, but this approach focuses on the TCP handshaking mechanism, not semantic protocol translation. S. Luo et al. exhibited IP translation to NDN in three different layers: Internet, TCP, and Application. The author named the IP packet using a unique hierarchical naming scheme. By simulating it using CCNx for running the FTP server and client, the authors showed that the translation mechanism was feasible. However, the translation scheme in [5] uses a non-standard interest packet to tell the consumer. As a result, it causes the packet overhead to NDN protocol. Moreover, the pre-stored data required an enormous NDN CS size for this scheme.

Similar to S. Luo, Rafaei, et al. in [7] did translation by using a gateway that consists of four components: Gateway Daemon for accepting packet, Gateway Configuration File for naming and configure the packet, Traffic Handler (TH) for forwarding the packet, and a Command Line Interface (CLI) for monitoring. However, each IP packet translation process in [7] is based on a full matched-packet filter that requires extra regex computation as overhead and shows potential security issues such as data privacy since reading the data payload to form a prefix name.

The dual-channel translation gateway focuses on building a translation gateway that is strict to the driven-pull semantics in the NDN family protocol without an extra overhead in communication transactions such as special interest. Our objective is to bring NDN semantics into IP protocol so the translation process can be done effectively. Moreover, an asymmetric name resolution table with two independent tables is introduced to provide the host-to-name binding flexibility.

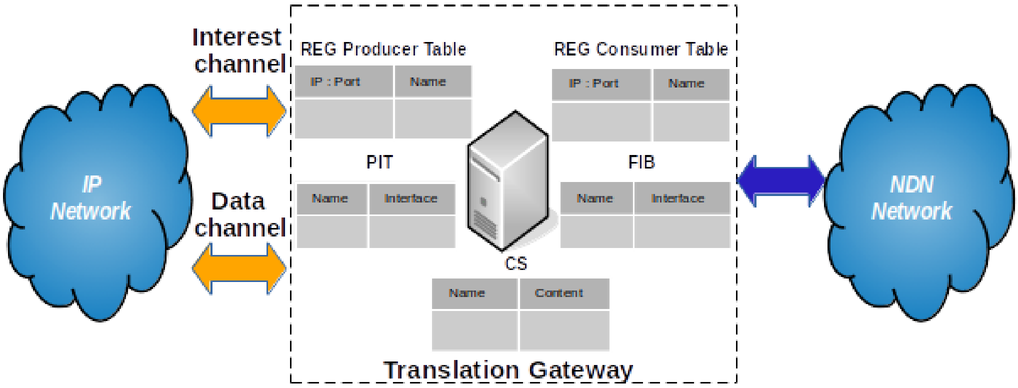

Fig. 1 Dual channel translation gateway architecture

3. The dual channel translation gateway

This section describes the architecture of the dual-channel translation gateway. The components of the gateway are illustrated briefly to give a general idea about the proposed architecture as seen in figure 1.

The components of the dual-channel translation gateway can be explained as follow:

- **The IP interest channel**

The IP interest channel is formed by a static IP address dedicated to send or receive an IP interest-like packet between the IP node and gateway. The static IP address for the data channel is known in advance by all IP nodes in the network.

- **The IP data channel**

The IP data channel also consists of a static IP address devoted to sending or receiving an IP data-like packet between the IP node and gateway. The static IP address assigned to the data channel is also acknowledged in advance by all IP nodes in the network.

- **Register table**

The register table gives information about the prefix name associated with the combination of IP address and port number. The table consists of two different independent tables, namely the producer table and the consumer table. The producer table is used when the IP node becomes a producer. On the other hand, the consumer table is used when the IP node becomes a consumer. The register table mechanism is like the Domain Name System (DNS) in the IP network with some extensions.

Besides those mentioned components above, the dual-channel IP-to-NDN translation gateway is also completed with the native components of the NDN router such as Content Store (CS), Pending Interest Table (PIT), Forwarding Information Based (FIB).

4. Throughput analysis model

Let τ denote the total processing time at the gateway for translation. The throughput is the function of delay so the throughput can be defined as $Throughput = f(\tau)$. Hence, the smaller the τ, the higher is the throughput. We identify that the formula of τ is different for each process and scenario. We define t_c as the searching time in CS when the cache is hit, t_r as the name look-up time in REG when the cache is hit, t_{cn} as the searching time in CS when the cache is miss, t_{i_p} as the interest parsing processing time, t_{i_s} as the interest serialization processing time, t_{d_p} as the data parsing processing time, t_{d_s} as the data packet serialization processing time, t_{ip_c} as the IP packet construction processing time, and t_{ip_d} as the IP packet deconstruction processing time.

Moreover, we also define t_{pr1} as the producer response time after receiving an interest in case of IP becoming producer and NDN becoming consumer (scenario 1), and t_{pr2} as the producer response time after receiving an interest in case of IP becoming consumer and NDN becoming producer (scenario 2).

In the case of scenario 1, the value of τ can be described by

$$\tau = \begin{cases} t_c + t_{i_p} (\equiv \tau_1) & \text{if CS hit} \\ t_{cn} + t_{i_p} + t_{ip_c} + t_{ip_d} + t_{d_s} + t_{pr1} (\equiv \tau_2) & \text{if CS miss} \end{cases}$$

However, in the case of scenario 2, the value of τ can be defined by

$$\tau = \begin{cases} t_c + t_r + t_{i_p} + t_{ip_c} + t_{ip_d} + t_{d_p} (\equiv \tau_3) & \text{if CS hit} \\ t_{cn} + t_r + t_{i_s} + t_{ip_c} + t_{ip_d} + t_{d_p} + t_{pr2} (\equiv \tau_4) & \text{if CS miss} \end{cases}$$

If the hit ratio in CS is β, and the number of interest packets send in a second is N, the average processing time, $\overline{\tau}$, can be obtained by

$$\overline{\tau} = (\beta\tau_1 + (1 - \beta)\tau_2)N, \tag{1}$$

and for scenario 2, $\overline{\tau}$ can be given by

$$\overline{\tau} = (\beta\tau_3 + (1-\beta)\tau_4)N. \qquad (2)$$

The value of β can be approximated by calculating the hit ratio of each prefix name, $P(i)$, from the total prefix name M. The formula can be expressed by

$$\beta = \sum_{i=1}^{M} P(i)\beta(i) \qquad (3)$$

The approximation of hit ratio could be calculated by using Che's approximation that was originally proposed by Che et al. [17]. Then, the hit ratio for a particular prefix name $\beta(i)$ could be estimated by

$$\beta(i) = 1 - e^{-q(i)t_c} \qquad (4)$$

The symbol $q(i)$ is the ratio of requests for content i, and t_c is the unique root of the equation

$$C = \sum_{i=1}^{M}(1 - e^{-q(i)t_c}) \qquad (5)$$

where C is the storage capacity of CS since we use LRU as a replacement algorithm in our translation gateway.

5. Numerical evaluation

The emulation uses a virtual box with 4 guest Operating systems (OS) and Python 3.7 as the network programming language. The emulation topology derives from the network at Figure 1.

Tab. 1 Setting values of main parameters

Paramater	Value
Number of content items	1000
Cache-size	10
REG-size	1000
Zipf parameter (α)	0.7 - 1.5
Interest rate	10000 interests/s

The major parameter values of the emulator are shown in Table 1. The interest packets were generated asynchronously about ten thousand packets in second to measure the throughput. The value of α affects the number of the popular prefix names emitted by the consumer. The smaller α creates the uniform prefix names, but the larger α creates the diverse prefix names. Furthermore, the emulation testbed was assumed that all prefix names are available in both REG producer and consumer tables.

For setting values of processing time components, we use two values, namely median and mean, taken from five hundred thousand packets in the emulation testbed. Table 2 shows the median and mean value of each processing time component.

Tab. 2 The values of processing time

Paramater	Median	Mean
t_c	1 μs	1.1 μs
t_{cn}	1 μs	0.9 μs
t_r	1 μs	1.2 μs
t_{i_p}	40 μs	47.05 μs
t_{i_s}	88 μs	122.5 μs
t_{d_p}	45 μs	65.4 μs
t_{d_s}	130 μs	188.8 μs
t_{ip_c}	4 μs	8.5 μs
t_{ip_d}	5 μs	9.5 μs
t_{pr1}	228 μs	255.3 μs
t_{pr2}	284 μs	346.2 μs

5.1 Hit ratio

The hit ratio β is an accumulation from M prefix names as shown in equation 3. We derived the equation 4 and 5 to achieve the analysis value of hit ratio β. Furthermore, we measured the emulation CS hit ratio by injecting some script in our emulator. Figure 2 shows the comparison of CS hit ratio between analysis and emulation results.

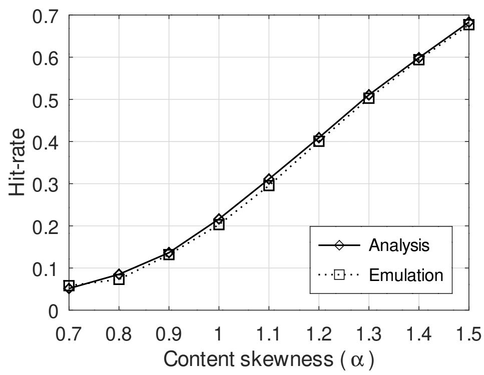

Fig. 2 Dual channel translation gateway architecture

The comparison gap between the analytical approach and emulation is about 0.01%, as seen in Figure 2. This result indicates that our model performance is accurate in predicting the CS hit ratio in the translation gateway.

5.2 Processing time distribution

The value of each processing time is required to calculate the average total processing time. Therefore, the distribution of processing time is necessary to illustrate the range and deviation of the captured processing time. We use two values, median and mean, taken from five hundred thousand packets from our emulation testbed. Table 2 shows the median and mean value of each processing time.

Figure 3 shows the probability distribution of τ_1 and τ_2 in scenario 1. As we can see, the distribution of τ_1 is narrower compared to τ_2 since the number of components of τ_1 is fewer than τ_2. However, the range of τ_1 still reaches 1000 times between the min and max value. Moreover, the time distribution of τ_2 is shifted to the right from τ_1 in scale of

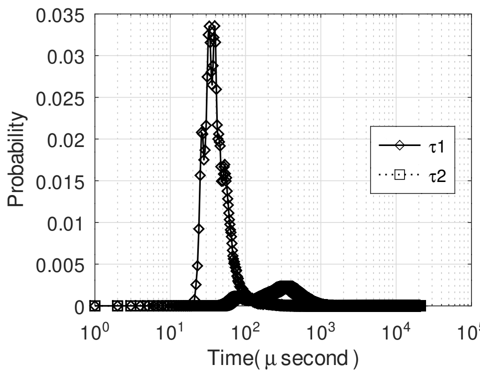

Fig. 3 Processing time distribution in scenario 1

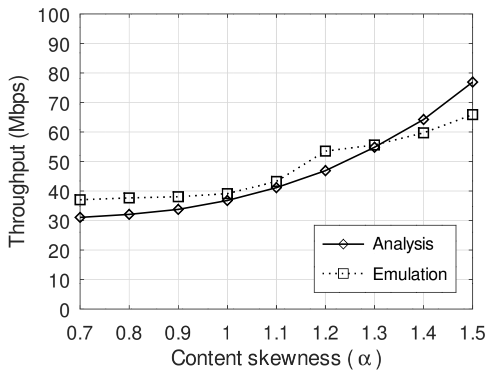

Fig. 5 Throughput analysis in scenario 1 by using median

100 μ seconds. It is also clear that the processing time distribution of τ_1 and τ_2 are skewed to the right where the mean value is greater than the median value.

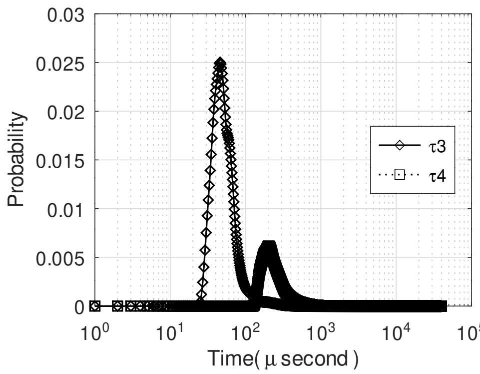

Fig. 4 Processing time distribution in scenario 2

Figure 4 shows the probability distribution of τ_3 and τ_4 in scenario 2. The time distribution of τ_4 is shifted to the right from τ_3 about 100 μ seconds. The shifting is caused by an additional component like t_{pr2}. The range for both processing times, τ_3 and τ_4, is almost similar, about 1000 times from the minimum and max value. The processing time distribution of τ_3 and τ_4 are skewed to the right, similar to scenario 1.

5. 3 Throughput

The throughput analysis uses equation 1 and 2 to obtain the average processing time in case of CS hit and CS miss. This processing time can be converted to the throughput by using a simple comparison between the best-case scenario and the calculation analysis for requesting 10000 interest within a second. Figure 5 shows the throughput comparison between emulation and analysis by using the median value in the scenario of IP producer and NDN consumer (scenario 1). Alternatively, Figure 6 depicts the throughput comparison between emulation and analysis by using mean value in scenario 1.

The findings demonstrate that, in general, median value throughput estimation exceeded mean value estimation in scenario 1. However, the mean value estimation significantly improved when α was accelerated.

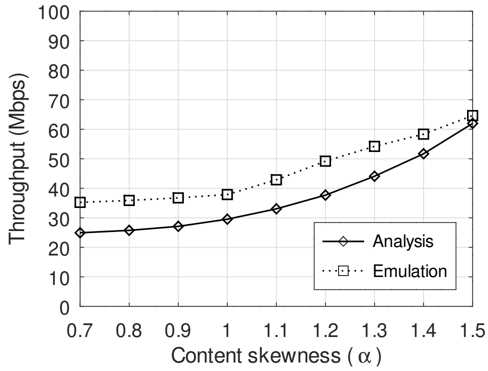

Fig. 6 Throughput analysis in scenario 1 by using mean

Figure 7 and 8 show the throughput comparison between emulation and analysis by using median and mean value, respectively, in scenario 2.

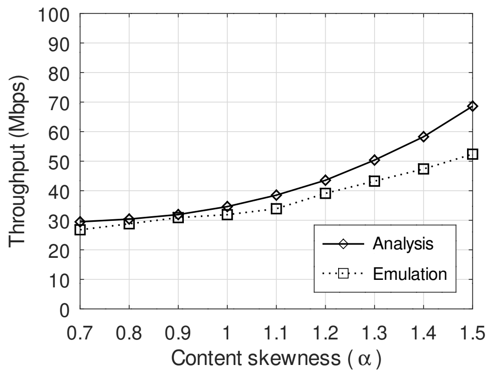

Fig. 7 Throughput analysis in scenario 2 by using median

In scenario 2, the data shows that the median estimation is better in the α lower than 1.0. Still, the mean estimation also shows moderate accuracy and tends to improve in the higher α.

The throughput gap between analysis and emulation is relatively smaller when α is lower than 1.0 in the median, but it is higher when α is larger than 1.0 for both scenarios. On the contrary, the throughput gap is smaller as α increases

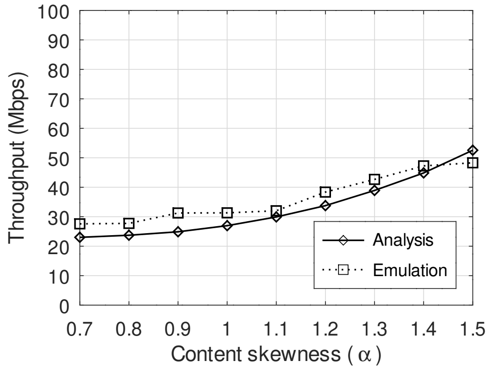

Fig. 8 Throughput analysis in scenario 2 by using mean

by using the mean value. The results have proofed that the mean value is effective in predicting the throughput when α is high. This phenomenon is caused by the higher deviation and processing time in τ_2 and τ_4 rather than τ_1 and τ_3. The higher α means that τ_1 and τ_3 are more dominant in calculation proportion than τ_2 and τ_4. On the other hand, the lower alpha causes τ_2 and τ_4 more dominant, so the lower alpha causes better throughput estimation by using the median value. It is because the wide and higher deviation can be minimized by using central values like the median.

6. Conclusion and Future work

The analysis shows that the parameter of hit ratio in CS is a critical variable for increasing the throughput significantly in the dual-channel IP-to-NDN translation gateway. Moreover, the results indicate that the processing time also affects the throughput. Therefore, faster hardware and software can also improve the throughput. Our throughput model estimation accuracy is about 70% on average, where median estimation is better than mean estimation in the range of α equal to and less than one. However, mean estimation has improved in the α greater than one for both scenario.

The investigation of time overhead in coding and Operating System (OS) switching context for each translation process in our dual-channel IP-to-NDN translation gateway are explored. Furthermore, the dynamic prefix name binding mechanism is also deeply examined in future work.

Ackonowledgement

This work was supported by JSPS KAKENHI Grant Number 18K11283 and 21H03437, as well as the Ministry of Religious Affairs of the Republic of Indonesia (MORA 5000 Doktor).

References

[1] B. Nour, F. Li, H. Khelifi, H. Moungla, and A. Ksentini," Coexistence of ICN and IP Networks: An NFV as a Service Approach," IEEE Global Communications Conference (Globecom), Waikoloa, USA, December 2019.

[2] F. Fahrianto and N. Kamiyama, "Comparison of Migration Approaches of ICN/NDN on IP Networks," ICIC-APTIKOM 2020, Online, Nov. 2020.

[3] F. Fahrianto and N. Kamiyama, "The Dual-Channel IP-to-NDN Translation Gateway," 2021 IEEE International Symposium on Local and Metropolitan Area Networks (LANMAN), 2021.

[4] G. Carofiglio, L. Muscariello, J. Auge, M. Papalini, M. Sardara, and A. Compagno," Enabling ICN in the Internet Protocol: Analysis and Evaluation of the Hybrid-ICN Architecture," Proceedings of the 6th ACM Conference on Information-Centric Networking, Macao, China, September 2019.

[5] S. Luo, S. Zhong, K. Lei, "IP/NDN: A Multi-Level Translation and Migration Mechanism," IEEE/IFIP Network Operations and Management Symposium, Taiwan, April 2018.

[6] H. Wu, J. Shi, Y. Wang, Y. Wang, G. Zhang, Y. Wang, B. Liu and B. Zhang, "On Incremental Deployment of Named Data Networking in Local Area Network," ACM/IEEE Symposium on Architectures for Networking and Communications Systems (ANCS), Beijing, China, May 2017.

[7] T. Refaei, J. Ma, S. Ha, and S. Liu, "Integrating IP and NDN through an Extensible IP-NDN Gateway," In Proceeding of ACM Information Centric Networking, Berlin Germany, September 2017 (ICN 2017).

[8] I. Moisennko and D. Oran, "TCP/ICN: Carrying TCP Over Content Centric and Name Data Networks," in ICN'16, Kyoto, Japan, September 2016.

[9] L. Zhang, A. Afanasyev, J. Burke, V. Jacobson, KC claffy, P. Crowley, C. Papadopoulos, L. Wang, and B. Zhang," Named Data Networking," ACM SIGCOMM Computer Communication Review, Vol. 4, July 2014.

[10] T. Liang, et al., NDNizing Existing Applications: Research Issues and Experiences, ACM ICN 2018.

[11] V. Jacobson, D. K. Smetters, J. D. Thornton, M. F. Plass, N. H. Briggs, R. L. Braynard, "Networking Named Content," CoNEXT 2009, Rome, December 2009.

[12] Z. Zhu, S. Wang, X. Yang, V. Jacobson, L. Zhang," ACT: Audio Conference Tool Over Named Data Networking," ICN 2011, Ontario, Canada, August 2011.

[13] V. Jacobson, D. K. Smetters, N. H. Briggs, M. F. Plass, P. Stewart, J. D. Thornton, R. L. Braynard," VoCCN: Voice-over Content-Centric Networks," ReArch'09, Rome, Italy December 2009.

[14] G. Xylomenos, Y. Thomas, X. Vasilakos et al., " IP Over ICN Goes Live, " in European Conference on Networks and Communications, June 2018.

[15] Jeff Thompson, Peter Gusev, and Jeff Burke., "NDN-CNL: A Hierarchical Namespace API for Named Data Networking," In Proceedings of the 6th ACM Conference on Information-Centric Networking. Association for Computing Machinery, New York, NY, USA, 2019.

[16] J. Hong, T. You and Y. Hong, "Name resolution service for CCN," International Conference on Information and Communication Technology Convergence (ICTC), Jeju, Korea (South), 2017.

[17] H. Che, Y. Tung, and Z. Wang, " Hierarchical Web Caching Systems:Modeling, Design and Experimental Results, " IEEE J. Selected Areas of Commun., vol.20, no.7, pp.1305-1314, Sep. 2002.

[18] Xinyu Ma, UCLA, CA, USA. python-ndn(2019) [Online]. Available: https://zjkmxy.github.io/projects/python-ndn.